Full citation: Yu-Kai Lin, Wen-Chieh Lin, and Chieh-Chih Wang, “Asynchronous State Estimation of Simultaneous Ego-motion Estimation and Multiple Object Tracking for LiDAR-Inertial Odometry,” *2023 International Conference on Robotics and Automation (ICRA)*, pp. 1–7, May. 2023.

Asynchronous State Estimation of Simultaneous Ego-motion Estimation and Multiple Object Tracking for LiDAR-Inertial Odometry

Yu-Kai Lin[1], Wen-Chieh Lin[1], and Chieh-Chih Wang[2]

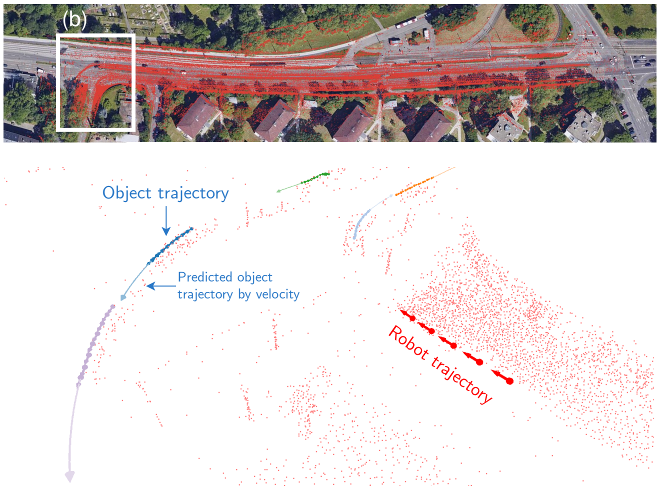
***Abstract*— We propose LiDAR-Inertial Odometry via Simultaneous EGo-motion estimation and Multiple Object Tracking (LIO-SEGMOT), an optimization-based odometry approach targeted for dynamic environments. LIO-SEGMOT is formulated as a state estimation approach with asynchronous state update of the odometry and the object tracking. That is, LIO-SEGMOT can provide continuous object tracking results while preserving the keyframe selection mechanism in the odometry system. Meanwhile, a hierarchical criterion is designed to properly couple odometry and object tracking, preventing system instability due to poor detections. We compare LIO-SEGMOT against the baseline model LIO-SAM, a state-of-the-art LIO approach, under dynamic environments of the KITTI raw dataset and the self-collected Hsinchu dataset. The former experiment shows that LIO-SEGMOT obtains an average improvement 1.61% and 5.41% of odometry accuracy in terms of absolute translational and rotational trajectory errors. The latter experiment also indicates that LIO-SEGMOT obtains an average improvement 6.97% and 4.21% of odometry accuracy.**

***Index Terms*— Autonomous driving, SLAM, odometry, multiple object tracking.**

I. Introduction

Simultaneous Localization and Mapping (SLAM) in dynamic environments is challenging because it assumes that surrounding scenes are stationary, but this assumption is usually violated in reality. Therefore, SLAM approaches are required to handle dynamic objects in real world applications to prevent estimation failure [1]. Meanwhile, Multiple Object Tracking (MOT) is essential to recognize surrounding dynamic information in many applications, such as robot navigation and autonomous driving. Integrating both components in real world applications becomes an important task to perceive robot states and the surrounding dynamic object motions in an unknown and time-varying environment [2].

The interaction of odometry and object tracking in dynamic environments is increasingly investigated in recent years. There have been mainly two strategies to integrate object tracking tasks in SLAM or odometry systems. One strategy performs a loosely-coupled optimization of odometry and object tracking, in which independent tracking results are used to refine odometry results for restricted purposes, such as scale recovery in visual odometry [4]–[6]. The other strategy views both odometry and object tracking tasks as a joint and tightly-coupled estimation problem. It derives a large and complex optimization problem in which all static and dynamic measurements would affect both odometry and object tracking results at the same time, while reliable dynamic measurements are claimed to advance the accuracy of odometry in dynamic environments [7], [8]. A perception problem that simultaneously estimates localization, static landmarks, and dynamic object movements is called Simultaneous Localization, Mapping, and Moving Object Tracking (SLAMMOT) [9]. A subproblem that does not estimate static landmarks is called Simultaneous EGo-motion estimation and Multiple Object Tracking (SEGMOT), and we focus on the SEGMOT problem in this paper.



(b) Trajectories of the robot and the surrounding objects.

Fig. 1: Results of LIO-SEGMOT under the KITTI raw sequence `0926-0014` [3]. The point cloud map in (a) is generated by the odometry result, and its bird’s eye view is aligned with the satellite image. A close up view of the white box in (a) is shown as (b), where the red trajectory is the robot trajectory, and trajectories encoded in different colors are surrounding objects’ trajectories. Colored arrows represent objects’ velocities.

In most odometry approaches, the keyframe selection mechanism is widely used for computational efficiency [10]–[12]. That is, only partial image or LiDAR frames are selected in the backend estimation processes. Therefore, all LiDAR information in non-keyframes are discarded from the odometry, and the robot state only updates in keyframes. However, object-related measurements in non-keyframes are also ignored when using the mechanism, resulting in decreasing accuracies of odometry and object tracking [2], [7]. We notice that robot states and surrounding objects’ states are formulated to be synchronous in most previous approaches, in which the factor graph optimization-based methods should not have this restriction.

In this paper, we propose LIO-SEGMOT, an approach

[1] Yu-Kai Lin and Wen-Chieh Lin are with the Institute of Multimedia Engineering, National Yang Ming Chiao Tung University, Hsinchu, Taiwan `yukai.cs09g@nctu.edu.tw`, `wclin@cs.nctu.edu.tw`

[2] Chieh-Chih Wang is with the Department of Electrical and Computer Engineering, National Yang Ming Chiao Tung University, and with the Mechanical and Mechatronics Systems Research Laboratories, Industrial Technology Research Institute, Hsinchu, Taiwan `bobwang@ieee.org`

to perform LiDAR-Inertial Odometry (LIO) via Simultaneous EGo-Motion estimation and Multiple Object Tracking (SEGMOT) in dynamic environments. The LIO-SEGMOT estimation problem is formulated as a nonlinear factor graph, where the subgraph representing the odometry component inherits LIO-SAM [12], a state-of-the-art LIO approach, and the other subgraphs representing the object tracking component comes from a variation of FG-3DMOT [13]. We formulate a novel asynchronous factor graph architecture that robot states and surrounding objects' states are asynchronous, which allows us to perceive complete objects' trajectories when using the keyframe selection mechanism. In addition, we introduce a hierarchical coupling condition for LiDAR object detections to avoid the system instability due to poor detections, where a similar idea was first introduced in the visual-based approach proposed by Liu *et al.* [7].

We evaluate the proposed method under the KITTI raw dataset [3] and the self-collected Hsinchu dataset. Sequences of both datasets within dynamic environments are chosen for evaluation, in which the Hsinchu dataset is a real world dataset containing many surrounding dynamic objects. Figure 1 shows a sample result of LIO-SEGMOT on a KITTI raw sequence, including the robot trajectory, objects' trajectories, and their velocities. Experimental results reveal that LIO-SEGMOT presents in average 1.61% and 5.41% better than LIO-SAM in terms of translational and rotational absolute trajectory errors (ATE) in the KITTI raw dataset, as well as in average 6.97% and 4.21% better than LIO-SAM in the Hsinchu dataset. Compared to the synchronous version, asynchronous factor graph formulation for LIO-SEGMOT provides continuous object tracking results, and presents in average more accurate than the synchronous version in both odometry and object tracking results. In summary, LIO-SEGMOT is shown to present accurate odometry results and continuous object tracking results in real world dynamic environments, in which robot states and surrounding objects' states are asynchronously estimated over time.

II. Related Work

A. Odometry with Loosely-Coupled Object Tracking

Odometry approaches with loosely-coupled object tracking perform two separate systems that estimate odometry and multiple object tracking independently [2], [5], [6], [14]–[16]. In particular, Lim *et al.* [5] simultaneously estimate camera trajectory and track a single person in a loosely-coupled way, where the absolute scale of the trajectory is recovered with a given height of the tracked human. Huang *et al.* introduce ClusterSLAM [16] for a backend of stereo visual SLAM and estimation of rigid body motion in dynamic environments, in which surrounding moving objects are represented as clusters of feature points with hierarchical agglomerative clustering [17], and their trajectories are revised with decoupled factor graph optimization. Following up on ClusterSLAM, ClusterVO [2] performs multi-level probabilistic association and a heterogeneous conditional random field cluster assignment for online clustering. In addition, a double-track frame management for sliding window-based state estimation is employed to preserve object tracking performance under the keyframe selection mechanism. Compared to the double-track frame management in ClusterVO [2], the proposed asynchronous state estimation approach does not need to constantly modify the factor graph architecture in previous timestamps, which avoids additional computational cost to maintain junction trees of incremental inference approaches.

B. Odometry with Tightly-Coupled Object Tracking

The main objective of odometry approaches with tightly-coupled object tracking is to develop a unified optimization problem that both tasks are jointly estimated and mutually affect each other during the optimization [4], [7]–[9], [18]–[22]. In **visual-based approaches**, Yang and Scherer [4] propose CubeSLAM to cooperate with monocular visual SLAM and object detection within a multi-view object SLAM in a tightly-coupled way, where objects are claimed to provide scale constraints to increase the accuracy of camera trajectory. Bescos *et al.* propose DynaSLAM II [21] to formulate a stereo and RGB-D visual SLAM that tightly integrates multiple object tracking in terms of factor graphs with instance segmentation and ORB features [23]. Liu *et al.* [7] introduce a probabilistic framework to allow objects to be tightly- or loosely-coupled to odometry in terms of detection and estimation uncertainty, and presents its feasibility in visual odometry. Compared to the proposed hierarchical criterion for tightly-coupled and loosely-coupled detection factors, Liu *et al.* [7] uses 2-D feature points to classify good or bad objects, whereas the proposed method uses 3-D object detections to check consistencies of object states and detections within the spatial domain. Recently, AirDOS [8] presents a dynamic SLAM approach which jointly estimates camera poses, object motions, and object structures within factor graphs by considering the rigidity and motion consistency of articulated objects.

As for **LiDAR-based approaches**, Wang *et al.* [9] firstly demonstrate feasibility of SLAMMOT in 2-D cases with laser scanners. The closely related work to the proposed method is DL-SLOT [22], which is recently proposed by Tian *et al.* to present a dynamic 3-D LiDAR SLAM with object tracking. However, DL-SLOT lacks the ability to refine the robot trajectory and object trajectories globally due to sliding window-based local optimization. Furthermore, DL-SLOT employs the tightly-coupled formulation of odometry and object tracking, which breaks the sparsity and robustness of the system that are also mentioned in [16] and [7], respectively. In contrast, the proposed method allows global refinement of all trajectories under an incremental inference framework. In addition, the sparsity and stability are improved with the proposed coupling strategy.

III. Proposed Method

In this section, we introduce the proposed LIO-SEGMOT that simultaneously estimates the robot trajectory and surrounding objects' trajectories. The odometry part of LIO-SEGMOT inherits the factor graph architecture of LIO-

SAM [12], since we can smoothly extend LIO-SAM [12] to a novel architecture that jointly performs robot odometry and object tracking within factor graphs. The object tracking part of LIO-SEGMOT uses LiDAR-based 3-D object detection approaches, and we propose a novel measurement model, called the mock detection measurement model, which allows the robot state and objects' states to be estimated asynchronously over time.

In general, a conservative data association criterion in the factor graph will degenerate the tracking capability, while an aggressive criterion will cause optimization instability of tracking quality due to poor detections. To mitigate the issue, LIO-SEGMOT uses a hierarchical criterion to determine the coupling of odometry and object tracking.

A. LiDAR Inertial Odometry via Smoothing and Mapping

LIO-SAM [12] provides a tightly-coupled LiDAR inertial odometry architecture with incremental inference for smoothing and mapping. It is formulated as a pose graph optimization problem in terms of factor graphs that contain robot states, IMU preintegration factors, LiDAR odometry factors, GPS factors, and loop closure factors. GPS factors and loop closure factors are optional in the framework and are skipped in this paper for the sake of conciseness.

Raw IMU measurements are used to compensate point clouds and to compute the robot motion during a LiDAR scan [24]. Meanwhile, a mechanism for selecting keyframes is introduced to improve computational efficiency. Each keyframe corresponds to a robot pose on the factor graph, and there are an IMU preintegration factor [25] and a LiDAR odometry factor connected to any two consecutive poses, in which the latter one is provided by LOAM-based scan matching [26] of a keyframe and a local map that is constructed from a fixed number of previous keyframes. The robot trajectory is incrementally updated and optimized with iSAM2 [27] when a robot pose and related factors are newly added to the factor graph. Furthermore, the updated trajectory is used to compute the temporal bias of IMU measurements.

B. Factor Graph Formulation in LIO-SEGMOT

Figure 2 shows the LIO-SEGMOT factor graph with asynchronous odometry and object tracking. The factor graph formulation consists of variable nodes and factor nodes, in which variable nodes contain states of the robot and its surrounding objects, and factor nodes provide odometry and object tracking constraints for optimization. In variable nodes, $\boldsymbol{x}_t \in SE(3)$ represents the robot pose, $\boldsymbol{x}_{t,i} \in SE(3)$ represents the i-th object's pose, and $\boldsymbol{v}_{t,i} \in SE(3)$ represents the i-th object's linear and angular velocities at t. As for factor nodes, apart from LiDAR odometry factors and IMU preintegration factors in LIO-SAM, we also introduce detection factors, constant velocity factors, and smooth movement factors for object tracking. From the perspective of optimization, detection factors constrain the relative transformation between robot poses and objects' poses, constant velocity factors constrain the consistency of two consecutive velocities of an object, and smooth movement factors constrain object movement between two consecutive poses of an object with its velocity.

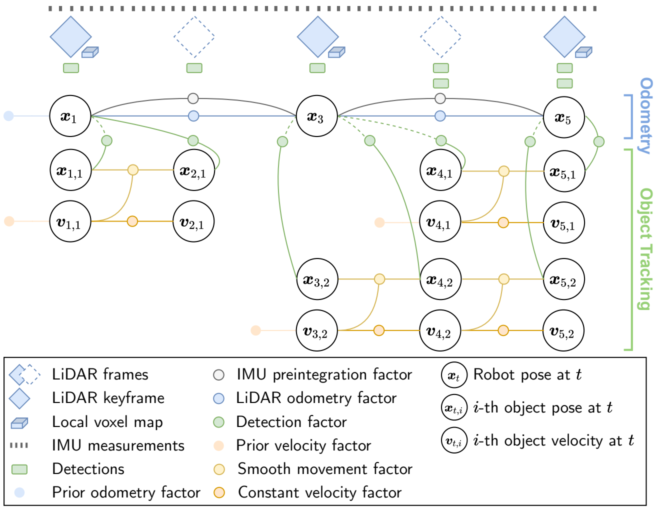
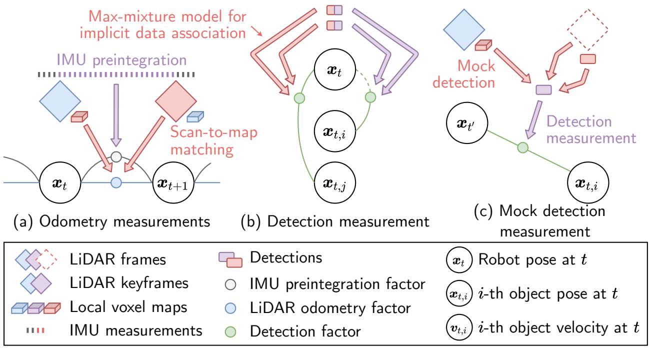


Fig. 2: A factor graph of LIO-SEGMOT for odometry and object tracking.



(a) Odometry measurements (b) Detection measurement (c) Mock detection measurement

Fig. 3: Measurement models in LIO-SEGMOT. Odometry measurements in (a) contain the LiDAR measurement and the IMU measurement. Detection measurements in (b) illustrate an implicit data association of two objects with both tightly-coupled and loosely-coupled detection factors. The mock detection measurement in (c) presents a novel measurement model that objects' states at non-keyframe timestamps can be properly estimated.

There are four measurement models used in LIO-SEGMOT, namely the LiDAR measurement model, the IMU measurement model, the detection measurement model, and the mock detection measurement model. In the odometry part of the system, LIO-SEGMOT adopts the LiDAR measurement model and the IMU measurement model introduced in LIO-SAM [12], as shown in Figure 3(a). For the object tracking part of the system, we design the detection measurement model that follows a max-mixture model for implicit data association between objects and detection measurements (as shown in Figure 3(b)), leading the assignment problem to be jointly optimized during the optimization. The idea is referred to FG-3DMOT [13] and the similar idea is also mentioned in MH-iSAM2 [28]. In addition, detection factors have tightly- and loosely-coupled forms, where the adjacent variable nodes of the tightly-coupled form are a robot pose and an object pose, while the only adjacent variable node of the loosely-coupled form is an object pose. The tightly- and loosely-coupled forms are distinguished in factor graphs by solid and dashed lines respectively, as shown in Figure 3(b). The major benefit to having both forms on a factor graph is to prevent system instability due to poor detections [7].

$$\begin{aligned}\min_{\mathcal{X}} & \underbrace{\sum_{(\boldsymbol{x}_t,\boldsymbol{x}_{t+1})\in\mathcal{O}} \left\| f_{\mathrm{O}}(\boldsymbol{x}_t,\boldsymbol{x}_{t+1}) \right\|^2_{\Sigma_{\mathrm{O}}}}_{\text{LiDAR odometry error}} + \underbrace{\sum_{(\boldsymbol{x}_t,\boldsymbol{x}_{t+1})\in\mathcal{I}} \left\| f_{\mathrm{I}}(\boldsymbol{x}_t,\boldsymbol{x}_{t+1}) \right\|^2_{\Sigma_{\mathrm{I}}}}_{\text{IMU preintegration error}} + \underbrace{\sum_{(\boldsymbol{x}_{t'},\boldsymbol{x}_{t,i})\in\mathcal{D}_{\mathrm{LC}}} \left\| f_{\mathrm{LC}}(\boldsymbol{x}_{t,i}|\boldsymbol{x}_{t'}) \right\|^2_{\Sigma_{\mathrm{LC}}}}_{\text{loosely-coupled object detection error}} + \underbrace{\sum_{(\boldsymbol{x}_{t'},\boldsymbol{x}_{t,i})\in\mathcal{D}_{\mathrm{TC}}} \left\| f_{\mathrm{TC}}(\boldsymbol{x}_{t'},\boldsymbol{x}_{t,i}) \right\|^2_{\Sigma_{\mathrm{TC}}}}_{\text{tightly-coupled object detection error}} \\ & + \underbrace{\sum_{(\boldsymbol{v}_{t,i},\boldsymbol{v}_{t+1,i})\in\mathcal{C}} \left\| f_{\mathrm{C}}(\boldsymbol{v}_{t,i},\boldsymbol{v}_{t+1,i}) \right\|^2_{\Sigma_{\mathrm{C}}}}_{\text{object constant velocity error}} + \underbrace{\sum_{(\boldsymbol{x}_{t,i},\boldsymbol{x}_{t+1,i},\boldsymbol{v}_{t,i})\in\mathcal{M}} \left\| f_{\mathrm{M}}(\boldsymbol{x}_{t,i},\boldsymbol{x}_{t+1,i},\boldsymbol{v}_{t,i}) \right\|^2_{\Sigma_{\mathrm{M}}}}_{\text{object smooth movement error}} + \underbrace{\left\| \operatorname{Log}(\boldsymbol{x}_0) \right\|^2_{\Sigma_{\mathrm{P}}}}_{\text{prior robot pose error}} + \underbrace{\sum_{\boldsymbol{v}_{t,i}\in\mathcal{V}} \left\| \operatorname{Log}(\boldsymbol{v}_{t,i}) \right\|^2_{\Sigma_{\mathrm{V}}}}_{\text{prior object velocity error}},\end{aligned} \tag{1}$$

To deal with detection constraints for object states at non keyframes (e.g., $\boldsymbol{x}_{2,1}$, $\boldsymbol{x}_{4,1}$, and $\boldsymbol{x}_{4,2}$ in Figure 2), the mock detection measurement model is newly introduced in LIO-SEGMOT. The main idea is to apply data fusion of LiDAR scans and object detections so that the detection factors in this case can be associated with the latest robot state. In practice, the coordinate of object detections is transformed from the non-keyframe to the latest keyframe. Figure 3(c) illustrates the mock detection measurement model for a detection at $t > t'$ that a mock detection is generated by the original detection and the relative transformation of the robot states at t and t', where the transformation is provided by the LOAM-based scan matching of the LiDAR frame at t and the local voxel map at t'. The mock detection measurement model not only lets the asynchronous state estimation problem be well-defined, but allows LIO-SEGMOT to perceive complete objects' states.

Different from the constant linear velocity model used in FG-3DMOT [13], we adopt the constant linear and angular velocity model (CLAV) to express the object dynamics to provide smooth translational and rotational movements of objects. In LIO-SEGMOT, constant velocity factors provide a soft constraint that velocities at two consecutive timestamps should be similar; smooth movement factors adopt CLAV to constrain object movement between two consecutive object states based on the estimated velocities.

The optimization problem of LIO-SEGMOT is formulated by Eq. (1), where $\mathcal{X}$ is the set of all variables, $\mathcal{O}$, $\mathcal{I}$, $\mathcal{D}_{\mathrm{LC}}$, $\mathcal{D}_{\mathrm{TC}}$, $\mathcal{C}$, and $\mathcal{M}$ are the sets of LiDAR odometry factors, IMU preintegration factors, loosely- and tightly-coupled detection factors (respectively), constant velocity factors, and smooth movement factors in terms of their adjacent variable nodes, respectively, and $\operatorname{Log}: SE(3) \to \mathbb{R}^6$ is the capitalized $SE(3)$ logarithm map [29]. The corresponding measurement functions of these factors are $f_{\mathrm{O}}, f_{\mathrm{I}}, f_{\mathrm{LC}}, f_{\mathrm{TC}}, f_{\mathrm{C}}: SE(3)^2 \to \mathbb{R}^6$, and $f_{\mathrm{M}}: SE(3)^3 \to \mathbb{R}^6$. Meanwhile, the corresponding covariance matrices are Σ_{O}, Σ_{I}, Σ_{LC}, Σ_{TC}, Σ_{C}, $\Sigma_{\mathrm{M}} \in \mathbb{R}^{6\times 6}$. The optimization problem also requires prior information on the robot pose and all object velocities with respect to the covariance matrices $\Sigma_{\mathrm{P}} \in \mathbb{R}^{6\times 6}$ and $\Sigma_{\mathrm{V}} \in \mathbb{R}^{6\times 6}$. The mathematical formulations of $f_{\mathrm{LC}}, f_{\mathrm{TC}}, f_{\mathrm{C}}$ and f_{M} are provided in the following sections:

1) Tightly- and Loosely-Coupled Detection Factors: The detection factors adopt an equally weighted Gaussian mixture model (GMM) with a maximum mixture approximation [30], [31]. Given the pose of the i-th object $\boldsymbol{x}_{t,i} \in SE(3)$ at t, the latest robot pose $\boldsymbol{x}_{t'} \in SE(3)$ at $t' \leq t$, and a finite non-empty set of (mock, if $t' < t$) detection measurements $\mathcal{Z}_t \subsetneq SE(3)$, the measurement function for the general detection factor is written as

$$f_{\mathrm{D}}(\boldsymbol{x}_{t'}, \boldsymbol{x}_{t,i}|\mathcal{Z}_t) = \operatorname{Log}(\boldsymbol{z}^{-1}\boldsymbol{x}_{t'}^{-1}\boldsymbol{x}_{t,i}), \tag{2}$$

where

$$\boldsymbol{z} = \underset{\boldsymbol{y}\in\mathcal{Z}_t}{\arg\min} \| \operatorname{Log}(\boldsymbol{y}^{-1}\boldsymbol{x}_{t'}^{-1}\boldsymbol{x}_{t,i})\|^2_{\Sigma_{\mathrm{D}}}. \tag{3}$$

The proposed method provides two types of detection factors, in which the tightly-coupled detection factor,

$$f_{\mathrm{TC}}(\boldsymbol{x}_{t'}, \boldsymbol{x}_{t,i}|\mathcal{Z}_t) = f_{\mathrm{D}}(\boldsymbol{x}_{t'}, \boldsymbol{x}_{t,i}|\mathcal{Z}_t), \tag{4}$$

views the pose of the robot $\boldsymbol{x}_{t'}$ and the pose of the object $\boldsymbol{x}_{t,i}$ as variables to be updated in the factor graph. On the other hand, the loosely-coupled detection factor,

$$f_{\mathrm{LC}}(\boldsymbol{x}_{t,i}|\boldsymbol{x}_{t'}, \mathcal{Z}_t) = f_{\mathrm{D}}(\boldsymbol{x}_{t'}, \boldsymbol{x}_{t,i}|\mathcal{Z}_t), \tag{5}$$

views the pose of the object $\boldsymbol{x}_{t,i}$ as a variable, but views the pose of the robot $\boldsymbol{x}_{t'}$ as a constant.

2) Constant Velocity Factors: The proposed method assumes that each object moves with a constant velocity in a short period of time. For a single object, the measurement function of the constant velocity factor with two consecutive velocity variables $\boldsymbol{v}_{t,i}, \boldsymbol{v}_{t+1,i} \in SE(3)$ is given by

$$f_{\mathrm{C}}(\boldsymbol{v}_{t,i}, \boldsymbol{v}_{t+1,i}) = \operatorname{Log}(\boldsymbol{v}_{t+1,i}^{-1}\boldsymbol{v}_{t,i}), \tag{6}$$

and the factor is only used for two consecutive timestamps.

3) Smooth Movement Factors: Smooth movement factors are used to constrain the motion behavior of objects. Following CLAV, the measurement function is given by

$$\begin{aligned} & f_{\mathrm{M}}(\boldsymbol{x}_{t,i}, \boldsymbol{x}_{t+1,i}, \boldsymbol{v}_{t,i}|\delta_{t,t+1}) \\ & \qquad = \operatorname{Log}\left(\boldsymbol{x}_{t+1,i}^{-1}\boldsymbol{x}_{t,i}\operatorname{Exp}(\delta_{t,t+1}\operatorname{Log}(\boldsymbol{v}_{t,i}))\right), \end{aligned} \tag{7}$$

where $\boldsymbol{x}_{t,i}, \boldsymbol{x}_{t+1,i} \in SE(3)$ are the i-th object's poses at timestamps t and $t+1$ respectively, $\delta_{t,t+1}$ is the time interval between timestamps t and $t+1$, and $\operatorname{Exp}: \mathbb{R}^6 \to SE(3)$ is the capitalized $SE(3)$ exponential map [29].

C. Hierarchical Criterion for Coupling and Tracking

To overcome the uncertainty of object detections, a hierarchical criterion for LiDAR object detections is proposed in LIO-SEGMOT to progressively make the following decisions when a new detection $\boldsymbol{z} \in SE(3)$ is coming into the system:

(Q1) Does the detection belong to any existing object $\boldsymbol{x}_{t,i}$?

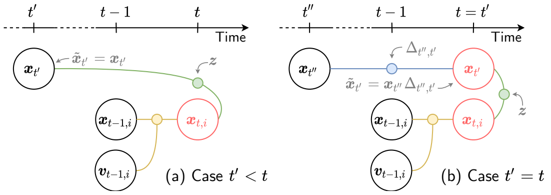
(Q2) If (Q1) holds, does $\boldsymbol{z}$ follows the i-th object's motion?

(Q3) If (Q1) and (Q2) hold, should the tightly-coupled detection factor be applied?

The first two questions (Q1) and (Q2) are determined by using the Mahalanobis distance of the error vector,

$$\left\| \operatorname{Log}(\boldsymbol{z}^{-1}\tilde{\boldsymbol{x}}_{t'}^{-1}\tilde{\boldsymbol{x}}_{t,i}) \right\|_{\Sigma} \leq \varepsilon, \tag{8}$$

with given covariance matrices $\Sigma \in \{\Sigma_{\mathrm{Q}_1}, \Sigma_{\mathrm{Q}_2}\} \subsetneq \mathbb{R}^{6\times 6}$ and a threshold $\varepsilon > 0$, where $\tilde{\boldsymbol{x}}_{t'}$ and $\tilde{\boldsymbol{x}}_{t,i}$ are initial estimations



(a) Case $t' < t$ (b) Case $t' = t$

Fig. 4: A graphical view of relationship between the robot and the i-th object. Red variables are new states that do not have previous estimates.

of the robot state $\boldsymbol{x}_{t'}$ at t' and the i-th object pose $\boldsymbol{x}_{t,i}$ at t based on current measurements and factor graph, i.e.,

$$\tilde{\boldsymbol{x}}_{t'} = \begin{cases} \boldsymbol{x}_{t'} & \text{if } t' < t, \\ \boldsymbol{x}_{t''}\Delta_{t'',t'} & \text{if } t' = t, \end{cases} \tag{9}$$

$$\tilde{\boldsymbol{x}}_{t,i} = \boldsymbol{x}_{t-1,i}\operatorname{Exp}(\delta_{t-1,t}\operatorname{Log}(\boldsymbol{v}_{t-1,i})), \tag{10}$$

where $\boldsymbol{x}_{t''}$ is the latest existing robot state at t'' in the factor graph, and $\Delta_{t'',t'}$ is the relative transformation of $\boldsymbol{x}_{t''}$ and $\boldsymbol{x}_{t'}$ computed by scan matching, as shown in Figure 4. We also assume that $\Sigma_{\mathrm{Q}_2} - \Sigma_{\mathrm{Q}_1}$ is positive semidefinite (i.e., $\Sigma_{\mathrm{Q}_2} - \Sigma_{\mathrm{Q}_1} \succeq 0$) to prevent ambiguity of the hierarchical criterion that (Q2) holds but (Q1) does not hold.

Two spatial information-based tests are conducted to determine (Q3), which are the detection constraint and the velocity constraint. The former focuses on the consistency between predicted object poses and detections, and the latter focuses on the consistency of previous velocities. In practice:

(Q3-1) (Detection constraint) Eq. (8) holds with another given covariance matrix $\Sigma_{\mathrm{Q}_{3,1}}$ that satisfies $\Sigma_{\mathrm{Q}_{3,1}} - \Sigma_{\mathrm{Q}_2} \succeq 0$.

(Q3-2) (Velocity constraint) The variance of velocities is small enough. That is,

$$\frac{1}{N}\sum_{s=1}^{N}\left\| \operatorname{Log}(\boldsymbol{v}_{t-s,i}) - \operatorname{Log}(\bar{\boldsymbol{v}}_{t,i}) \right\|^2_{\Sigma_{\mathrm{Q}_{3,2}}} \leq \varepsilon \tag{11}$$

with a given covariance matrix $\Sigma_{\mathrm{Q}_{3,2}}$, where N is the fixed number of previous velocities of object states and $\bar{\boldsymbol{v}}_{t,i} \in SE(3)$ is the mean of the N previous velocities.

If (Q1) holds for the detection $\boldsymbol{z}$ and the corresponding i-th object, the new state of the i-th object along with a loosely-coupled detection factor would be added to the factor graph. Furthermore, if (Q2) holds, a constant velocity factor and a smooth movement factor would be also added to the factor graph. Finally, if (Q3) holds, the loosely-coupled detection factor would be replaced with a tightly-coupled detection factor. It means that the i-th object are regarded as a reliable object that are suitable to refine the odometry.

IV. Experiments

We evaluate LIO-SEGMOT in two real world datasets, the KITTI raw dataset [3] and the self-collected Hsinchu dataset, to the proposed method in dynamic environments[1]. We use the translational and the rotational absolute trajectory errors, $\mathrm{ATE_T}$ (meter) and $\mathrm{ATE_R}$ (degree), in terms of root-mean-square errors (RMSE), as the evaluation metrics for odometry [32]. All experiments are conducted on a desktop computer with an Intel i7-11700 CPU, 32GB RAM, and a GeForce RTX 3070 8GB graphical card.

TABLE I: Absolute robot trajectory errors and computational times (CT) in *second* of LIO-SAM and LIO-SEGMOT under the KITTI raw dataset. Bold text means the best result, and underlined text means the suboptimal result. Results of both synchronous and asynchronous LIO-SEGMOT in `0926-0013` and `0926-0032` are same as each LiDAR frame is considered to be keyframe; i.e., their factor graphs are identical.

Sequence	LIO-SAM			LIO-SEGMOT (synchronous)			LIO-SEGMOT (asynchronous)		
	$\mathrm{ATE_T}$	$\mathrm{ATE_R}$	CT	$\mathrm{ATE_T}$	$\mathrm{ATE_R}$	CT	$\mathrm{ATE_T}$	$\mathrm{ATE_R}$	CT
`0926-0009`	0.526	0.966	**95.8**	**0.503**	0.956	116.0	0.512	**0.951**	135.7
`0926-0013`	0.217	1.116	**33.9**	**0.214**	**0.954**	36.5	**0.214**	**0.954**	36.5
`0926-0014`	0.601	5.062	**75.4**	**0.570**	**4.871**	80.4	0.573	4.974	81.3
`0926-0015`	0.362	6.764	**75.3**	0.385	6.148	82.3	**0.336**	**5.239**	83.0
`0926-0032`	**1.021**	14.505	**108.5**	1.124	**14.019**	116.5	1.124	**14.019**	116.5
`0926-0051`	0.284	3.130	**98.3**	0.300	**3.109**	102.4	**0.266**	3.310	111.1
`0926-0101`	5.826	**75.108**	**226.8**	**5.803**	75.180	245.6	5.875	75.148	248.1

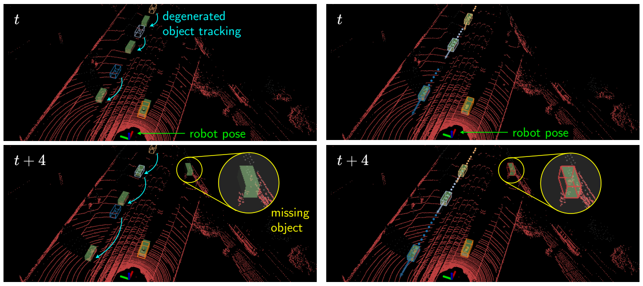
A. KITTI Raw Dataset

Data Collection. The KITTI raw dataset [3] is selected instead of other KITTI datasets as it provides complete LiDAR scans and IMU data. Sequences including moving objects are chosen for evaluation. The 64-beams LiDAR scanner (Velodyne HDL-64E) provides averagely 10 scans per second, and the IMU sensor (OxTS RT3003) provides averagely 100 records per second. SE-SSD [33] with the pre-trained model is adopted to detect 3-D objects in point clouds.

Experimental Results. Table I shows the odometry results on the KITTI raw dataset. The asynchronous LIO-SEGMOT are on average 1.61% and 5.41% better than LIO-SAM in translational and rotational errors, respectively. On the other hand, the synchronous LIO-SEGMOT does not present comparable odometry accuracies to the asynchronous version, and does not output better odometry accuracies than LIO-SAM in terms of translational absolute trajectory errors. To investigate the phenomenon, we compare object tracking results of the synchronous LIO-SEGMOT and the asynchronous LIO-SEGMOT quantitatively. Table II lists the tracking results in average translational and rotational relative pose error, $\mathrm{RPE_T}$ and $\mathrm{RPE_R}$, in terms of RMSE of all surrounding moving object trajectories. We observe that the asynchronous LIO-SEGMOT outperforms the synchronous LIO-SEGMOT in trajectory errors of moving objects. This indicates that the asynchronous LIO-SEGMOT benefits from complete detection measurements in object tracking. In addition, the asynchronous LIO-SEGMOT receives more object detection cues for both odometry and object tracking tasks, leading to promisingly robust and accurate robot trajectories and surrounding objects' trajectories in dynamic environments.

We also perform a qualitative analysis for object tracking results of the synchronous LIO-SEGMOT and the asynchronous LIO-SEGMOT. Figure 5 shows a visual comparison between the synchronous and asynchronous versions of LIO-SEGMOT in the KITTI sequence `0926-0101`. The robot is stationary in the first second, and there are several surrounding dynamic vehicles. Since the synchronous version does not update new object states when incoming a non-keyframe LiDAR scan, the states of dynamic objects are not updated accordingly. It results in degenerated object tracking results, and the object association could fail in the next state. By contrast, the asynchronous version can provide

[1] Code and hyperparameter setting are available at `https://github.com/StephLin/LIO-SEGMOT`.



(a) Synchronous version. (b) Asynchronous version.

Fig. 5: Visual comparison of object tracking results for synchronous and asynchronous LIO-SEGMOT under the KITTI sequence `0926-0101`. Green solid cuboids are detection measurements, and 3-D bounding boxes in different colors are different tracked objects. Colored dots and arrows are objects' trajectories and velocities, respectively. One can see that in timestamps t and $t+4$, asynchrinous estimation results produce continuous and accurate object poses and velocities, while synchrinous estimation results do not perform object tracking as the robot is stationary.

TABLE II: Average translational (m/frame) and rotational (deg/frame) relative pose errors of all surrounding moving objects under the KITTI raw dataset. The KITTI sequence `0926-0101` is omitted since there is no ground truth tracking result for evaluation.

Sequence (#moving objects)	LIO-SEGMOT (synchronous)		LIO-SEGMOT (asynchronous)	
	RPE_T	RPE_R	RPE_T	RPE_R
`0926-0009` (4)	0.142	2.151	**0.128**	**1.870**
`0926-0013` (2)	**0.156**	**1.673**	**0.156**	**1.673**
`0926-0014` (16)	0.174	1.766	**0.153**	**1.475**
`0926-0015` (7)	**0.156**	1.555	**0.156**	**1.531**
`0926-0032` (12)	**0.196**	**1.725**	**0.196**	**1.725**
`0926-0051` (8)	0.165	1.607	**0.157**	**1.505**

continuous object tracking results over time even though there is only one single robot state in the factor graph.

B. Hsinchu Dataset

Data Collection. The Hsinchu dataset is collected at Guangfu road in Hsinchu city, which is a high dynamic urban scene crowded by cars and motorcycles. Figure 6 shows an overview of the sequence in the Hsinchu dataset, and we can see that there are numerous vehicles around the data collection car. The 32-beams LiDAR scanner (Velodyne VLP-32C) provides on average 10 scans per second, and the IMU sensor (Xsens-MTI-G-710) provides on average 100 records per second. PointPillars [34] with a pre-trained model [35] is used in this experiment as the detection model performs better adaptation on the Hsinchu dataset. The ground truth robot trajectory is computed with tactical-grade IMU, GNSS, LiDAR, and wheel encoder.

Experimental Results. Table III shows the experiment results in terms of translational and rotational absolute robot trajectory errors in the Hsinchu dataset. Compared to LIO-SAM, the odometry result of the asynchronous LIO-SEGMOT presents in an average improvement 6.97% and 4.21% in translational and rotational trajectory errors, respectively. It shows the feasibility of the asynchronous LIO-SEGMOT to provide accurate odometry with informative dynamic objects in highly dynamic environments. We also quantitatively compare the object tracking results of the synchronous LIO-SEGMOT and the asynchronous LIO-SEGMOT. Table IV lists the tracking results in average translational and rotational relative pose error of surrounding moving objects in the first 3/8 part of the GuangfuRoad sequence. Similar to the previous results, we observe that the asynchronous LIO-SEGMOT also outperforms the synchronous LIO-SEGMOT in tracking accuracy. Compared to the synchronous version, the asynchronous LIO-SEGMOT demonstrates the importance of utilizing complete detection information in both odometry and object tracking systems.

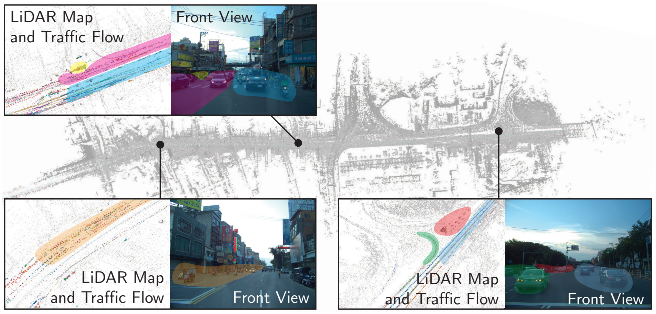


Fig. 6: LiDAR map and the traffic flow of the GuangfuRoad sequence in the Hsinchu dataset. Gray points are point clouds, and colored dots are objects' trajectories. Three close-up views with front views exhibit high dynamic scenes that crowded by vehicles.

TABLE III: Absolute robot trajectory errors and computational time (CT) in *sec* of LIO-SAM and LIO-SEGMOT under the Hsinchu dataset. Bold text indicates the best result, and underlined text means the suboptimal result.

Sequence	LIO-SAM			LIO-SEGMOT (synchronous)			LIO-SEGMOT (asynchronous)		
	ATE_T	ATE_R	CT	ATE_T	ATE_R	CT	ATE_T	ATE_R	CT
GuangfuRoad	1.204	3.349	**338.1**	1.537	**3.189**	1864.9	**1.120**	3.208	3392.9

TABLE IV: Average translational (m/frame) and rotational (deg/frame) relative pose errors of all moving objects in the Hsinchu dataset. The moving objects of the first 3/8 part of the sequence are labelled for evaluation.

Sequence (#moving objects)	LIO-SEGMOT (synchronous)		LIO-SEGMOT (asynchronous)	
	RPE_T	RPE_R	RPE_T	RPE_R
GuangfuRoad (66)	0.554	3.214	**0.503**	**2.833**

V. Conclusion

We present LIO-SEGMOT, a dynamic object-aware LiDAR-inertial odometry approach via simultaneous ego-motion estimation and multiple object tracking. We formulate a nonlinear factor graph to estimate LiDAR-inertial odometry with coupled dynamic object tracking, where the two subsystems are seamlessly integrated by introducing the mock detection measurement model. In addition, a hierarchical criterion for coupling and tracking is introduced to conquer the uncertainty of object detections. Real world experiment results indicate that LIO-SEGMOT presents a comparable or better accuracy than LIO-SAM in dynamic environments, in terms of absolute trajectory errors. Compared to the synchronous version, the asynchronous LIO-SEGMOT further provides continuous and accurate object tracking results.

In the future, we would like to speed up LIO-SEGMOT by optimizing the proposed factor graph with multi-robot iSAM2 [36]. Furthermore, we plan to extend LIO-SEGMOT to a factor graph-based mix-integer programming problem with MH-iSAM2 [28] so that it can explore global optimality among all combinatorial possibilities of detection coupling.

References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[2] J. Huang, S. Yang, T. Mu, and S. Hu, "Clustervo: Clustering moving instances and estimating visual odometry for self and surroundings," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 2165–2174.

[3] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*. IEEE Computer Society, 2012, pp. 3354–3361.

[4] S. Yang and S. A. Scherer, "Cubeslam: Monocular 3-d object SLAM," *IEEE Trans. Robotics*, vol. 35, no. 4, pp. 925–938, 2019.

[5] H. Lim and S. N. Sinha, "Monocular localization of a moving person onboard a quadrotor MAV," in *IEEE International Conference on Robotics and Automation, ICRA 2015, Seattle, WA, USA, 26-30 May, 2015*. IEEE, 2015, pp. 2182–2189.

[6] A. Kundu, K. M. Krishna, and C. V. Jawahar, "Realtime multibody visual SLAM with a smoothly moving monocular camera," in *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, D. N. Metaxas, L. Quan, A. Sanfeliu, and L. V. Gool, Eds. IEEE Computer Society, 2011, pp. 2080–2087.

[7] Y. Liu, J. Liu, Y. Hao, B. Deng, and Z. Meng, "A switching-coupled backend for simultaneous localization and dynamic object tracking," *IEEE Robotics Autom. Lett.*, vol. 6, no. 2, pp. 1296–1303, 2021.

[8] Y. Qiu, C. Wang, W. Wang, M. Henein, and S. A. Scherer, "Airdos: Dynamic SLAM benefits from articulated objects," in *International Conference on Robotics and Automation, ICRA 2022*. IEEE, 2022.

[9] C. Wang, C. E. Thorpe, S. Thrun, M. Hebert, and H. F. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *Int. J. Robotics Res.*, vol. 26, no. 9, pp. 889–916, 2007.

[10] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[11] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[12] T. Shan, B. J. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "LIO-SAM: tightly-coupled lidar inertial odometry via smoothing and mapping," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2020, Las Vegas, NV, USA, October 24, 2020 - January 24, 2021*. IEEE, 2020, pp. 5135–5142.

[13] J. Pöschmann, T. Pfeifer, and P. Protzel, "Factor graph based 3d multi-object tracking in point clouds," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2020, Las Vegas, NV, USA, October 24, 2020 - January 24, 2021*. IEEE, 2020, pp. 10 343–10 350.

[14] S. Song and M. Chandraker, "Joint SFM and detection cues for monocular 3d localization in road scenes," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2015, pp. 3734–3742.

[15] K. M. Judd, J. D. Gammell, and P. Newman, "Multimotion visual odometry (MVO): simultaneous estimation of camera and third-party motions," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, October 1-5, 2018*. IEEE, 2018, pp. 3949–3956.

[16] J. Huang, S. Yang, Z. Zhao, Y. Lai, and S. Hu, "Clusterslam: A SLAM backend for simultaneous rigid body clustering and motion estimation," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 5874–5883.

[17] R. R. Sokal, "A statistical method for evaluating systematic relationships." *Univ. Kansas, Sci. Bull.*, vol. 38, pp. 1409–1438, 1958.

[18] K. Lin and C. Wang, "Stereo-based simultaneous localization, mapping and moving object tracking," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, October 18-22, 2010, Taipei, Taiwan*. IEEE, 2010, pp. 3975–3980.

[19] M. Henein, J. Zhang, R. E. Mahony, and V. Ila, "Dynamic SLAM: the need for speed," in *2020 IEEE International Conference on Robotics and Automation, ICRA 2020, Paris, France, May 31 - August 31, 2020*. IEEE, 2020, pp. 2123–2129.

[20] J. Zhang, M. Henein, R. E. Mahony, and V. Ila, "VDO-SLAM: A visual dynamic object-aware SLAM system," *CoRR*, vol. abs/2005.11052, 2020.

[21] B. Bescós, C. Campos, J. D. Tardós, and J. Neira, "Dynaslam II: tightly-coupled multi-object tracking and SLAM," *IEEE Robotics Autom. Lett.*, vol. 6, no. 3, pp. 5191–5198, 2021.

[22] X. Tian, J. Zhao, and C. Ye, "DL-SLOT: dynamic lidar SLAM and object tracking based on graph optimization," *CoRR*, vol. abs/2202.11431, 2022.

[23] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, D. N. Metaxas, L. Quan, A. Sanfeliu, and L. V. Gool, Eds. IEEE Computer Society, 2011, pp. 2564–2571.

[24] C. L. Gentil, T. A. Vidal-Calleja, and S. Huang, "IN2LAMA: inertial lidar localisation and mapping," in *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*. IEEE, 2019, pp. 6388–6394.

[25] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation," in *Robotics: Science and Systems XI, Sapienza University of Rome, Rome, Italy, July 13-17, 2015*, L. E. Kavraki, D. Hsu, and J. Buchli, Eds., 2015.

[26] J. Zhang and S. Singh, "LOAM: lidar odometry and mapping in real-time," in *Robotics: Science and Systems X, University of California, Berkeley, USA, July 12-16, 2014*, D. Fox, L. E. Kavraki, and H. Kurniawati, Eds., 2014.

[27] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "isam2: Incremental smoothing and mapping using the bayes tree," *Int. J. Robotics Res.*, vol. 31, no. 2, pp. 216–235, 2012.

[28] M. Hsiao and M. Kaess, "Mh-isam2: Multi-hypothesis isam using bayes tree and hypo-tree," in *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*. IEEE, 2019, pp. 1274–1280.

[29] J. Solà, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," *CoRR*, vol. abs/1812.01537, 2018.

[30] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," in *Robotics: Science and Systems VIII, University of Sydney, Sydney, NSW, Australia, July 9-13, 2012*, N. Roy, P. Newman, and S. S. Srinivasa, Eds., 2012.

[31] T. Pfeifer and P. Protzel, "Expectation-maximization for adaptive mixture models in graph optimization," in *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*. IEEE, 2019, pp. 3151–3157.

[32] Z. Zhang and D. Scaramuzza, "A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, October 1-5, 2018*. IEEE, 2018, pp. 7244–7251.

[33] W. Zheng, W. Tang, L. Jiang, and C. Fu, "SE-SSD: Self-Ensembling Single-Stage Object Detector From Point Cloud," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 2021, pp. 14 494–14 503.

[34] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 12 697–12 705.

[35] Livox, "Livox detection v1.1," https://github.com/Livox-SDK/livox_detection, 2020.

[36] Y. Zhang, M. Hsiao, J. Dong, J. Engel, and F. Dellaert, "MR-iSAM2: Incremental smoothing and mapping with multi-root bayes tree for multi-robot SLAM," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2021, Prague, Czech Republic, September 27 - Oct. 1, 2021*. IEEE, 2021, pp. 8671–8678.